# CS534 - Machine Learning

# IA1 Competition

Vy Bui - 934370552

**Instructor: Dr. Xiaoli Fern**

The School of Electrical Engineering and Computer Science
Oregon State University

# BASELINE

The implementation from the Implementation Assignment 1, with some small modifications, was used as the starting point. The train data from **PA1_train1.csv** was split into **training_data** and **val_data**, which comprise 80% and 20% of the data respectively. The **test_data** is read from **PA1_test1.csv**. All of the data was then normalized using $\mu_{train}$ and $\sigma_{train}$ computed from **training_data**. Similar as the group IA1, date was split and ages since renovated was added to the feature list.

Experimented learning rates can be found in Table 1.

| Learning Rate | MSE |
|---|---|
| 1 | Diverge |
| 0.5 | Diverge |
| 0.15 | 4.367230 |
| 0.1 | 4.3682759056458 |
| 0.075 | 4.509987609852156 |
| 0.05 | 4.533113029839304 |
| 0.025 | 4.557692194914489 |
| 0.01 | 4.489928626263307 |
| 0.001 | 4.575720392043068 |
| 0.0001 | 10.808924189860564 |

Table 1: different learning rates and corresponding MSE

0.15 seems to yields the best model, named $model_1$, whose $MSE = 4.367230$ . $model_1$ is re-trained with all of the input data. Its features' importance can be found in Table 2.

| Feature | Weight |
|---|---|
| bias | 5.370881 |
| bedrooms | -0.293320 |
| bathrooms | 0.349722 |
| sqft_living | 0.765111 |
| sqft_lot | 0.057593 |
| floors | 0.018585 |
| waterfront | 1.447041 |
| view | 0.575169 |
| condition | 0.180739 |
| grade | 1.121869 |
| sqft_above | 0.743586 |
| sqft_basement | 0.176143 |
| yr_built | -0.914694 |
| zipcode | -0.276595 |
| lat | 0.839431 |
| long | -0.312246 |
| sqft_living15 | 0.139174 |
| sqft_lot15 | -0.093799 |
| month | 0.045941 |
| day | -0.054593 |
| year | 0.183123 |
| age_since_renovated | -0.161380 |

Table 2: $model_1$'s weights

# FEATURE EXPLORATION

Firstly, to gain some insights into the data, pandas was used to compute the correlations between input features and the target features (price), which are showned in Table 3.

| Input Features | Correlation with price |
|---|---|
| id | -0.014748 |
| bedrooms | 0.304994 |
| bathrooms | 0.524480 |
| sqft_living | 0.693156 |
| sqft_lot | 0.090327 |
| floors | 0.265757 |
| waterfront | 0.222654 |
| view | 0.392961 |
| condition | 0.051306 |
| grade | 0.671957 |
| sqft_above | 0.605777 |
| sqft_basement | 0.295117 |
| yr_built | 0.057532 |
| yr_renovated | 0.095046 |
| zipcode | -0.048750 |
| lat | 0.307248 |
| long | 0.025544 |
| sqft_living15 | 0.589190 |
| sqft_lot15 | 0.085476 |
| price | 1.000000 |

Table 3: correlations between price and input features

The top three features with highest correlation with "price" are "sqft_living", "grade", and "sqft_above", from highest to lowest. The correlation between "long" and "price" is quite small, which is reasonable because all of samples are houses located in Seattle, whose longitudes are very close to each other. This might indicate that "long" is not quite useful to predict the house price. However, training without using "long" results in worse models.

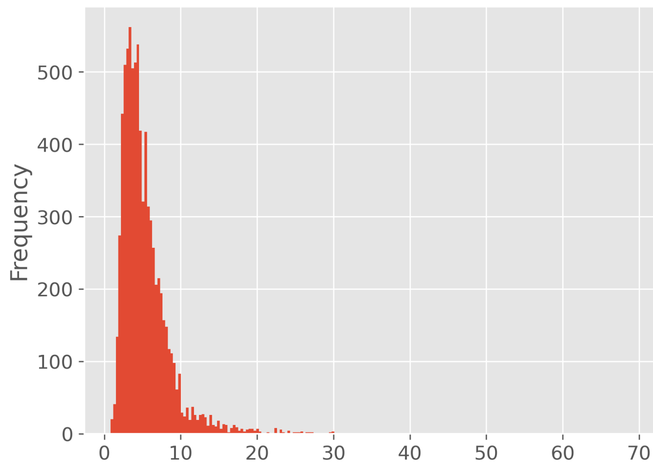


Figure 1: Histogram of price

The house prices mainly fall in the range from [0.82,14.4], with more than 97%, as showned by Figure 1. Suspecting that the outliers might negatively affect the model, I dropped the input data with prices $> 14.4$ to train $model_2$. It performs worse than $model_1$, having $MSE = 5.28$.

The correlations between price and preprocessed input features are shown in Table 4.

| Preprocessed Input Features | Correlation with price |
|---|---|
| bedrooms | 0.304994 |
| bathrooms | 0.524480 |
| sqft_living | 0.693156 |
| sqft_lot | 0.090327 |
| floors | 0.265757 |
| waterfront | 0.222654 |
| view | 0.392961 |
| condition | 0.051306 |
| grade | 0.671957 |
| sqft_above | 0.605777 |
| sqft_basement | 0.295117 |
| yr_built | 0.057532 |
| zipcode | -0.048750 |
| lat | 0.307248 |
| long | 0.025544 |
| sqft_living15 | 0.589190 |
| sqft_lot15 | 0.085476 |
| price | 1.000000 |
| month | -0.008468 |
| day | -0.024775 |
| year | 0.001692 |
| age_since_renovated | -0.099004 |

Table 4: correlations between price and preprocessed input features

# CONCLUSION

First, $model_1$ appears to be the best model with the MSE of 4.37. Second, using random data split into train and validation set can result in models with significantly different performance.